Letter to the Editor

Dear CrossTalk Editor,

Kevin Stamey opened the November 2005 issue with these remarks in his "From the Sponsor" column:

> ... other engineering design disciplines have been in place for centuries; however, software engineering is still relatively new. The discipline of software design has only been matured for a few decades. It wasn't until the 1960s that the first software products hit the marketplace ... Our dominant programming language, C++, didn't emerge until the 1980s ...

The relative newness of software engineering is often cited when explaining the frustrations of the ongoing software crisis. However, the fact that current practices have only been around for a few decades, is that really extraordinary? Is our phenomenal growth all that unique? And have all the other engineering disciplines really been around for centuries?

Aeronautical and aerospace engineering may not be as new as software engineering, but there are certainly not centuries of experience in those fields. Not long ago, most aircraft were propeller-driven, and we referred to the sound barrier. Electrical engineering cannot be considered a centuries-old discipline unless you start with Ben Franklin's kite-and-key experiments.

Many of software engineering's principal tools have indeed been in place for a relatively short time, but isn't that true of most engineering disciplines? Niels Bohr's simplistic model of the atom is less than 100 years old. Physicists are continually discovering new particles; researchers are only beginning to explore the possibilities of quantum computing. Huge advances have been made by materials scientists, meaning circuitry and silicon technologies have undergone several significant advances in a relatively short time.

Indeed, our newness presents some formidable challenges, and provides fodder for intense debate. But we ought to avoid emphasizing that this newness makes us unique, or that our needing to adapt to rapidly evolving technologies and standards is somehow exclusive. Such naiveté presents us as making excuses for our shortcomings rather than boldly confronting challenges.

The first transistor was fabricated in the 1940s, and the first rudimentary integrated circuits were fabricated in the 1950s, about the same time that early compilers came into being. Software engineers don't need more time for their field to mature; like others in technological and engineering fields, we are challenged to advance and progress in a disciplined yet rapid fashion to keep up with the monumental advances occurring in the world around us.

There are several aspects of software engineering that set us apart from other engineering disciplines. Most notably, our end product is tied to the virtual world, not the physical world. As such, our discipline is governed less by the laws of physics, and we don't rely on equations as fundamental, foundational truths. This makes it harder to build upon the previous work of theoreticians in a predictable way – something that I think better explains our slow maturation than our relative newness.

John Reisner
Air Force Institute of Technology
<jreisner@gimail.af.mil>